Defining and Validating Embedded Computer Software Requirements
Using the ECS, OTPM and IPFA

Dr. John H. Manley
Professor of Industrial Engineering
Director, Manufacturing Systems Engineering Program
University of Pittsburgh
jmanley@engrng.pitt.edu

1 Introduction

In the 1960s a new class of highly specialized digital computers began to evolve from the existing worlds of general purpose "automatic data processing" business machines and specialized analog computers. In 1973 this new computer type was first formally defined by the author for the US Air Force as an "embedded computer" since they were being engineered into so called "embedded computer systems" (ECS).[1] An ECS was defined as a stand-alone, real-time, semi-automated system such as the then-new B-1A strategic bomber aircraft. Since the early 1970s embedded computers have become ubiquitous and appear as integral parts of most manufactured products, from talking toys and automobiles to aerospace vehicles. They are also used extensively to integrate complex repetitive processes which represent the nervous systems of large-scale enterprises such as satellite launch centers and modern factories.[2]

This paper describes a field-tested, common-sense, macro-modeling solution to the now 25-year-old problem of how to cost-effectively define and validate certain critical embedded computer software requirements. The solution employs the author's Object Transformation Process Model (OTPM) [Figure 2] which can be used to macro-model any ECS architecture and comprehensively identify the minimum essential information (MEI) required by specified humans, machines, and computers involved in time-critical processes.[3] OTPM-based modeling results in: (1) the identification of explicit inputs and outputs required for each involved embedded computer program, and (2) the required timing for data arrival at and computer information departure from any embedded computer. This information is used to develop an improved requirement definition for the stored embedded computer programs needed to transform embedded computer input data sets into required output computer information to humans, control signals to machines, or computer data to other embedded computers. Finally, output timing and data volume specifications can be derived from this information and used to define bandwidth requirements for designing ECS supporting telecommunication systems.

2 ECS Computer Information Flow Analysis

All automated manufacturing lines, manned space vehicles, military aircraft, nuclear power plants, and similar real-time automated systems contain three control elements: (1) embedded computers, (2) humans-in-the-loop, and (3) certain electromechanical and/or analog devices, e.g., switches, sensors, and motors. The three elements are interconnected to make up the parent system and both individually and collectively require precisely-timed, highly-accurate, closed-loop control systems to ensure both human safety and high quality output products and/or services from their parent systems. Digital computers embedded in time- and safety-critical systems are especially difficult control elements to initially design, validate, and subsequently upgrade during parent systems reengineering projects, e.g., modernizing factories, and finding sources of year 2000 (Y2K) problem code in embedded processors.

Industrial engineers can assist computer systems and communication engineers design new and/or reengineer troublesome or obsolescent real-time ECS by ensuring that all control loops, both feed forward and feed back, are complete and efficient. The primary tool used for such work is a modified industrial engineering Process Flow Analysis (PFA) procedure called an Information Process Flow Analysis (IPFA).[4] The IPFA uses a combination of an ECS physical model [Figure 1] and the OTPM [Figure 2] as a conceptual framework to guide specialized analyses of complex, large-scale, embedded systems. When the IPFA is used to reengineer systems that are considered "troublesome" by management, the analysis objectives are twofold: (1) identify the minimum essential, necessary and sufficient information required for controlling any type of digital or analog computer embedded in large, complex, real-time automated systems, and (2) eliminate unnecessary and/or redundant information to and from embedded computers in order to improve the parent system's overall throughput and efficiency. What is different about IFPA is that it is focused on improving information flows—not material flows. Thus, the normal use of PFA by industrial engineers to eliminate waste from moving, storing, and unnecessary handling of physical material is of secondary importance to IPFA. The primary focus of IFPA is on ensuring that all control loops, both feed forward and feed back are complete, validated, and made as efficient as economically possible.

2.1 Embedded Computer System (ECS) Model Description

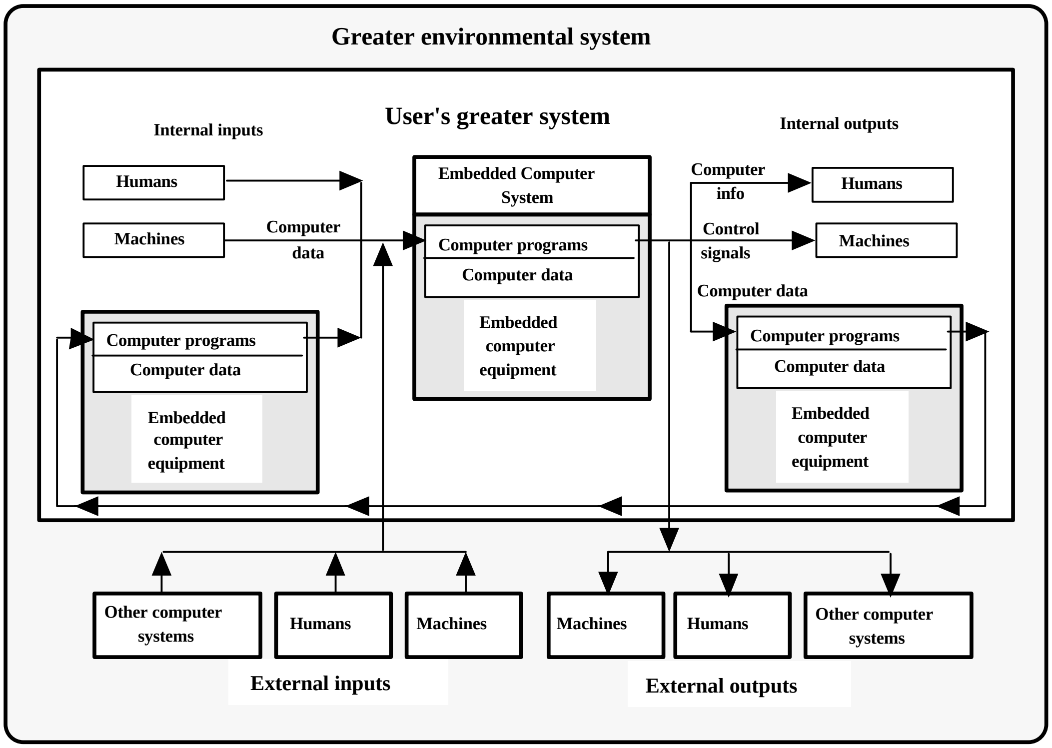

Figure 1: Embedded Computer System Model

The ECS model in Figure 1 has the following four important attributes:

1. The ECS describes *systems nested within systems*. An embedded computer system can be a single processor containing computer software (computer programs and computer data) which controls an electromechanical system, such as the flight controls of an aerospace vehicle, or desired pacing of an automated materials handling system. A "user's greater system" is composed of an integrated complex of humans, machines and multiple embedded computer systems that comprise, for example, a highly-automated aircraft, or an entire factory. A "greater environmental system," such as an airport that contains a control tower, aircraft maintenance facilities, etc. provides external inputs and receives outputs to support several "greater user systems" such as a fleet of aircraft. This nesting approach, or "onion skin model" can be continued upward to higher conceptual levels, such as multiple airports connected together by an air traffic control system, and so on.

2. The ECS emphasizes *three distinct sources of ECS input computer data*: (1) data generated by humans such as a single digital pulse from pushing a button, to a complex keyed or voice input data string, (2) analog or digital data generated by machines, such as a position signal from a servomechanism or the operation of a limit switch, and (3) digital data generated as output from another ECS within a user's greater system.

3. The ECS emphasizes *three distinct types of ECS outputs*: (1) computer-generated (human-understandable) information for humans, (2) computer-generated (machine-understandable) control signals for electromechanical machinery, and (3) computer-generated (computer program-understandable) output computer data from one ECS's computer program for another ECS's computer program as input data.

4. The ECS emphasizes that *all types of ECS inputs and outputs can coexist within a user's greater system, and also can originate from and terminate in the ECS's greater environmental system*.

Communication system issues addressed in the context of the ECS involve the physical and logical interconnections between embedded computers, humans, and machines that are integral to enterprise-wide manufacturing information systems. Today's manufacturing and aerospace systems engineers must not only understand the fundamentals of communication technology as it applies to embedded computer systems, but they must also know how to intelligently use this capability to install successful factory, office, and manned vehicle automation systems.

2.2 Object Transformation Process Model (OTPM) Description

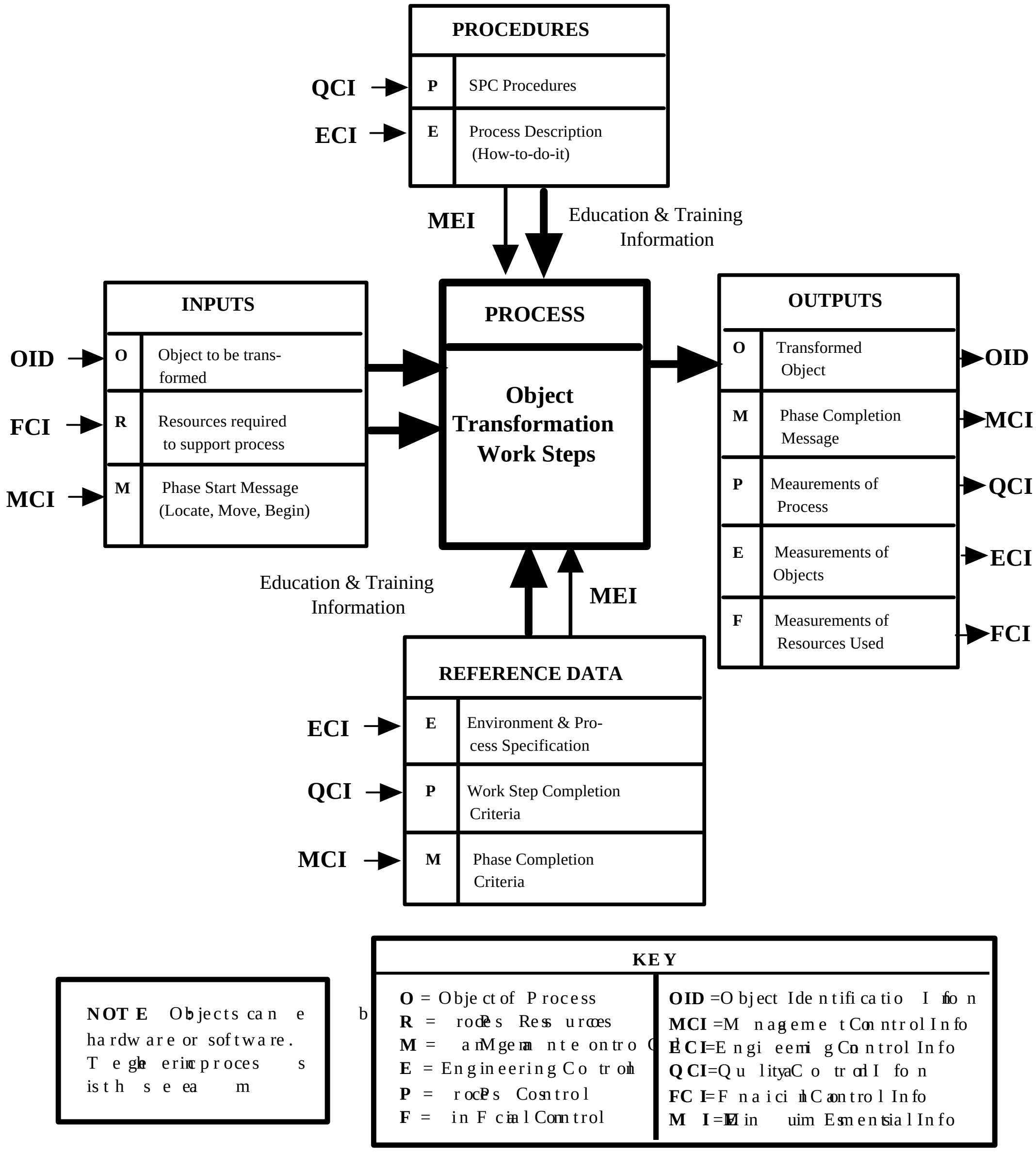

Figure 2: Object Transformation Process Model (OTPM)

The OTPM in Figure 2 describes a generic, arbitrarily-defined object transformation process (OTP) phase (which can be defined at any level of detail) and four information support components: inputs, outputs, procedures, and reference data. The physical input object is the output of a preceding transformation process phase. In the case of manufacturing, the starting point of a birth-to-death, repetitive manufacturing life cycle process considers the first objects to be transformed are crude oil or metal ore out of the ground; or biological materials from the surface such as tomatoes or peaches for transformation into food products. This first phase of the manufacturing process is defined as refining which produces raw materials or producer goods such as fuels, plastic, wood, metal, or specialized food items such as pastes or purees. In OTPM terms, refining represents the first phase of the manufacturing OTP life cycle. Therefore, the object transformation sequence from birth to death is as follows:

- Refining or raw material processing.
- Raw material production.
- Parts fabrication.
- Product assembly.
- Product modification, repair and/or re-manufacturing.

- Product disassembly for recycling and/or reuse.
- Product destruction and/or disposal.

The input object to each phase is, of course, any object that was transformed by a previous process step. In the case of intelligent product manufacturing, e.g., anti-lock braking systems for automobiles, all software-intensive component parts are treated in the same manner as hardware objects using the OTPM since all software eventually ends up embedded in a physical component, i.e., logic chips, ROM, "firmware," etc., or "modules" in the case of autos. Thus, in addition to a physical material transformation process, there also exists an *intellectual object transformation process* that, for example, transforms in a stepwise manner a human's mental concept of a computer program (software) into a physical read-only-memory (ROM) configuration that is eventually assembled into some "smart" product. Hence, the OTPM was designed to satisfy this manufacturing systems engineering need for using the same generic model to describe both physical and "intellectual object" transformations, namely hardware and software. The OTPM requires any manufacturing process phase or work package to contain four basic categories of information, each of which is an integral part of the OTPM. As indicated above, one relates to object identification, and three to process control.

- OID—Object identification information uniquely describes the object of interest. This information accompanies the object and changes as the object is progressively transformed.
- MCI—Management control information denotes both specific direction from transformation phase supervisors as input and reference data, as well as feedback output information that returns to the supervisors.
- ECI—Engineering control information comprises “how-to-do-it” process descriptions and process environment specifications as input, and transformed object measurements as feedback information to the engineers.
- QCI—Quality control information comprises work step completion criteria and statistical process control procedures as input information, and measurements of the transformation process as feedback output information to the quality control personnel.

2.3 Engineering Control Information Perspective

Traditional manufacturing engineering control connotes the process of ensuring that a transformed object meets or exceeds tolerances specified by an engineering design drawing. The OTPM uses the term "engineering" in a much broader sense. For example, if the object transformation process involves preparing an annual corporate tax return, the "engineer" who defines the how-to-do-it procedures in all likelihood would be a certified public accountant (CPA) trained in tax matters. The point is that the transformation process must be specified by an *expert* who, upon receiving detailed feedback information on what the transformed object looks like (a completed tax return), can readily identify errors and make rapid corrections to the methods and tools being used (machines or software) that perform the transformation. For specifying parameter settings on complex machine tools, this is clearly the responsibility of, e.g., mechanical, electrical, manufacturing or software engineers.

2.4 Quality Control Information Perspective

The term "quality" has many different meanings, especially in the context of a manufacturing enterprise. For OTPM purposes, quality is a primary attribute of not only every object that flows through an enterprise, but also the process itself. In particular, process stability (the ability to repetitively produce identical engineering results) is of major concern, as is the identification of object defects that cannot readily be controlled by engineering designs, e.g., those that result from human errors. Therefore, for any given level of quality of an object input into a subsequent transformation process phase or work step, the transformation process itself will determine the output level of object quality which has two components: (1) that which engineering can control, and (2) that which has to do with efficiency. For example, in one manufacturing plant, a particular fabricated part may continually conform to engineering specifications but takes twice as long to produce in another plant. The difference can be attributed to *process variability* (efficiency) since object variation is not an issue.

The author has arbitrarily decided to focus a highly desirable "continuous process improvement" function under the umbrella of quality control to distinguish it from traditional engineering quality control. The OTPM model also removes quality control from consideration as a testing and assurance function, in keeping with a very important principle that quality be built into products to prevent defects, i.e., “do it right the first time.”

Therefore, from a manufacturing information systems perspective, quality control information (QCI) is focused on making continuous process improvements in regard to both shortening process cycle times, and also reducing process variability by identifying new engineering and management controllable variables for incorporation into the engineering and management control systems. QCI is collected for all important object transformation processes using real-time statistical process control (SPC) techniques. QCI is fed back to a new paradigm quality control organization for appropriate analysis and action as part of enterprise-wide "total quality" initiatives.

3 Enterprise-wide Process Integration Through the OTPM

As shown in Figure 2, the OTPM elaborates each of the input, output, process, and control boxes by labeling information flows for objects [O], management [M], engineering [E], financial [F], and process (quality) control [P]. Resources required for a specific transformation as part of financial budget control is designated [R]. From a business perspective, no object transformation process can be undertaken without direction from management, since it represents a direct expenditure of material, human, and other resources. Also, no transformation can be considered complete unless information is generated and fed back to the manager who authorized the operation. Finally, for every controlled process, management must provide an explicit criteria for successful completion.

After completing their planning and organizing functions, managers direct manufacturing operations through an input (phase start message). They also control through the output (phase completion message) and reference data (phase completion criteria). By time-stamping input and output messages to and from any process, managers (or engineers) can determine any process end-to-end cycle time. Conformance to the reference criteria and procedures helps control the quality of each operation. For both human and machine processes, effective control is established because people (and machines) are expected to “perform as they are measured.”

The sheer simplicity of this model can be deceiving, since it can also integrate a number of processes within a group, department, division, or even an entire manufacturing plant into a single process. This can be accomplished by connecting outputs (feedback messages) from one object transformation process as the input to another process (direction message). To satisfy these integration requirements, we can theoretically combine the individual process models of an entire enterprise using their information flows as depicted by the OTPM using the same linking method. The main feature of this architectural building block is its *uniform four-dimensional interface for any process down to individual work steps*. Thus, complex *logical* process models having this configuration can be linked together in essentially the same manner that children build complex static structures out of Legos, and systems engineers design and construct dynamic worldwide telecommunication systems. The key to success in both cases are straightforward and understandable logical interfaces between the component parts. From the resulting logical models, we can construct viable *physical models* for controlling any business, engineering, or other control system. This is done using the ECS model referenced above in a similar manner to how it has been used by systems engineers for almost two decades to architect complex aerospace systems.

3.1 IPFA Link to Information System Requirements Specifications

Initial information requirements (or subsequent deficiencies) identified and documented through the IPFA must be translated into requirement specifications for engineering or reengineering supporting throughput, financial, business, engineering, and or total quality information systems in accordance with the OTPM conceptual framework. Note that the Structured Analysis and Design Technique (SADT) tool[5] that has been enhanced by the Air Force into IDEF (the Integrated Computer and Manufacturing ICAM DEFinition methodology)[6] appears on the surface to be similar to the OTPM IPFA methodology in regard to reengineering information systems that control real-time embedded manufacturing systems. However, the OTPM provides an added cross-functional higher level perspective of an enterprise that is not normally described by IDEF. In accordance with the IPFA methodology, traditional PFA provides $IDEF_0$ with task information, follow-up OTPM analysis provides $IDEF_1$ with what information is required, and $IDEF_2$ adds the missing element of when the information is required, as can be conceptualized from the ECS model.

Note also that IDEF, OTPM and IPFA are based on classical structured techniques, and are therefore compatible with information system software requirement specification development tools such as data flow diagrams, structure charts, entity-relationship diagrams, and most Computer Aided Software Engineering (CASE) tools. Other potential candidates for IPFA linkage also exist such as a state-based methodology using Requirements State Machine Language (RSML) which involves a graphical specification language that is both readable and reviewable by applications experts who are not computer scientists or mathematicians.[7] Note, however, that neither IDEF or RSML are *object-oriented* in the sense of emerging software development methodologies.[8] Therefore, one important subject of my university research is to find even more effective ways to link IPFA structured analysis results to object-oriented and/or other types of information system software requirement specification methods and tools.

In short, the OTPM and ECS conceptual models provide a framework to guide process flow analyses that can help industrial and manufacturing system engineers identify critical information errors of commission and omission during product and service manufacturing system reengineering projects. In addition, traditional industrial engineering PFA is enhanced by adding the requirement to analyze the necessity and sufficiency of object identification and process control information that supports physical and/or intellectual object transformation processes. Finally, IPFA results can be used to develop requirement specifications for system engineering or reengineering any type of product or service manufacturing information system.

4 Using the OTPM to Improve Designer, User and Constructor Mutual Understanding

The totality of OTPM analysis and modeling methodology outputs specify multifaceted and integrated real-time information systems that support end-to-end physical and/or intellectual object transformation processes, primarily products and/or services that satisfy customer's expressed requirements. According to Olle,[9] any design product resulting from the design activities should include specifications that are understandable to the acceptors. For example, these may be the set of *user acceptors* who are required to review and submit positive approval of the design work. The specifications may be also reviewed by a *constructor acceptor*, who will apply very different judgments from the user acceptor.

Since the OTPM methodology focuses almost exclusively on the user, achieving mutual understanding between OTPM designers and users should not pose any major problems. However, special attention must be paid to establishing effective communications between the OTPM information system engineer and the constructor acceptors, the latter being primarily information system hardware, software and communications professionals. One of the biggest barriers to good communication in this case is the highly specialized technical jargon that each party uses to communicate with each other—which is a continuing problem for information system professionals throughout the world. In this case, the solution to achieving mutual understanding is to prescribe an *interface language* made up of carefully chosen and defined words and phrases which can be mutually understood. Unfortunately, to the best of my knowledge there are no universal interface language standards to enhance mutual understanding among information system analysts, designers and constructors. Therefore, the concepts and terminology recommended by the IFIP are used to the maximum extent possible in the OTPM methodology specification to help make OTPM-based design products more easily understood by constructor acceptors. In this regard, key elements of the OTPM methodology are summarized in the tables below which compares it to some of the traditional methods mentioned above. Additional details can be found in the author’s book, *Rise Above the Rest: The Power of Superior Information, Knowledge, and Wisdom.*[10]

Process Orientation

OTPM	TRADITIONAL
End-to-end requirement-to-delivery product and service repetitive processes (manufacturing)	Data flows and their supporting processes
Cross-functional processes are required	Processes normally limited to selected functional areas
Enterprise-wide process integration promoted through the OTPM methodology	Process integration discouraged by traditional methodology (becomes too complex)
Optimize end-to-end process information requirements	Suboptimize process functions without regard for precedent or subsequent functions

Data Orientation

OTPM	TRADITIONAL
Minimum essential information (MEI) to support object transformations	All available information and relationships between data elements NOTE: This is especially troublesome with current enterprise requirements planning (ERP) methods
Data are categorized into management, financial, engineering, quality, and object identification	No standard method for categorizing data
Data are defined from a process definition	Data are cross-referenced to functional areas to ensure availability

Behavior Orientation

OTPM	TRADITIONAL
Transformation process corrective actions taken by humans, machines, or computers are triggered by means of comparisons of input with output MEI	Computerized tasks are triggered by human, computer or machine actions
OTPM specifies where and when MEI are needed to support human-to-human mutual understanding	Specifies user system events which trigger changes in data flow processes

Modeling Technique

OTPM	TRADITIONAL
Information Process Flow Analysis (IPFA) used to identify and model relationships of MEI needed to support any "manufacturing" process[11]	Required data elements identified by user(s) and information system analysts model their relationships
Embedded Computer System model used as a common framework for physical implementation of OTPM enterprise-wide architectures	No common interface model to link logical data flow models to physical architectural designs
MEI defined for intellectual and material object transformation process phases help determine physical location and type of databases required	Existing or planned database systems determine their physical location

5 OTPM Solution Field Experience and Results

Manufacturing industry sponsored field tests beginning in 1991 have clearly demonstrated that the OTPM and IPFA are highly useful for improving data identification, collection, analysis, feedback, and related manufacturing processes in ECS environments, especially for information system improvement. Most experiments have been conducted as integral parts of University of Pittsburgh Manufacturing Systems Engineering Program internships which required students to complete either a major capstone research project or formal M.S. thesis internship with their committees composed of both faculty and industry project supervisors.

1. The first major field test was conducted at the Packard Electric Division of General Motors Corporation in Warren, Ohio where an OTPM-modified PFA of a 320-machine plastic molding plant was used to identify and eliminate waste generated by ineffective communications in both business and engineering shop-floor semi-automated control systems. The improved information systems produced thesis-documented savings of over $1,800,000 per year.

2. The second major field test was conducted at AEG Westinghouse Transportation Services Inc. [currently ABB Adtranz] in West Mifflin, Pennsylvania that focused on a cycle time analysis of airport people mover manufacturing final assembly and testing. The OTPM-modified cycle time analysis uncovered the need for improved engineering control data. When such data were subsequently defined, collected and analyzed, the results led to making major changes to the final assembly process. This, in turn, generated an annual savings of over $600,000 and doubled the manufacturing plant’s overall production capacity

3. Another early field test was conducted at Bloom Engineering, Inc. that manufactures customized metallurgical furnace control systems, each of which is uniquely designed for a single customer need. The focus there was on improving the generation and control of product design information to identify and separate that which is unique to a job from that which can be reused on future jobs. A new engineering OTPM-based database was designed and programmed which significantly improved Bloom's front-end engineering design processes.

Since 1993, 80 practicing engineers who average 11 years of industrial experience have learned how to implement the methodology described in this paper by taking the author’s graduate course in manufacturing information systems reengineering. Most importantly, they have carried out 23 team projects to identify and solve ECS-based information system problems using the OTPM-based methodology in a variety of manufacturing and service organizations. In every project to date, the student teams were able to make significant improvements to existing automated processes, or plan information system solutions which were accepted by the target enterprises for subsequent implementation.

In addition to university research projects, the author has been teaching the methodology to other industrial organizations through short course offerings and also as part of his corporation’s educational services offerings to government and industry. A sample of the wide variety of information system problems that have been solved by practicing engineers (as MSEP student researchers) using the OTPM methodology are provided in the table below.

ORGANIZATION STUDENT RESEARCHER	INFORMATION PROBLEM SOLVED
Petro1eos de Venezuela (PDVSA) Hugo R. Vasquez-Tarbay	The OTPM was used to analyze the global RDC-generation process in Venezuela's nationalized oil company from the arrival of an RDC [purchase requisition in Spanish] at PDVSA Services, Inc. in Houston, Texas [PDVSA central purchasing agency in the US] to the selection of the panel of vendors for quotation. A relational database prototype was developed to unify 70 different procedures for RDC generation being used to gather information from PDVSA's country-wide operating divisions. This solution radically improved the quality of the selection of the computer codes by clients and subsidiaries.
Delphi Packard Electric Systems, General Motors Corporation Annette M. Pohlman	Applied the OTPM-based enterprise continuous improvement strategy to develop a measurement system for streamlining General Motors Corporation's multi-plant automobile power and signal distribution system product development process. The strategy was adopted for implementation by management.
Cornelius Architectural Products, Inc. Mark F. Rothert	Designed a standard coding system for communicating complex information relative to project material and components within a manufacturing organization to eliminate waste from interpersonal misunderstandings. The projects by Cornelius were one-of-a-kind, e.g., creating complex signage displays for Disneyland in Orlando, Florida which were exceedingly difficult to describe in understandable detail at the front end of the manufacturing process.
Superior Valve Company Tracy R. Shaffer	A total process perspective was used to define a model for assembly manufacturing cell design that incorporated product demand. Specifically, the model showed that once machine-component groupings are formed, the cell formation process can be further refined by sorting the product family by demand and grouping parts into replicated cells according to high and low to medium volume parts. The layout for each replicated cell was matched to the grouping's production demand and adequate cell utilization is now maintained by comparing demand to capacity for each replicated cell.
Delphi Packard Electric Systems, General Motors Corporation Daniel D. Gottfried	Used OTPM information process flow analysis (IPFA) to help define the minimum, relevant information systems requirements of a five-stage manufacturing process for producing automotive ignition cable. The problem solved was determining the root cause and implementing corrective actions to eliminate loose core cable defects from the extrusion of composite, high temperature core, rubber insulated ignition cable. "The use of object-oriented information systems facilitated more concise problem definition and timely discovery of causal factors that could be controlled or eliminated within the improved manufacturing system."
The Elliott Company Elizabeth A. Samstag	Used OTPM principles to analyze and improve the receiving function for a gas turbine power generator manufacturer from the delivery of materials from vendors and carriers, to initial receipt of materials, inspection, repackaging, and final placement of these materials in their warehouse locations.
Delphi Packard Electric Systems, General Motors Corporation John A. Sankovich	Taguchi methods were first used to define the impact of extrusion processing factors on fuselink cable processing. These factors were then optimized in order to achieve fuselink with the desired insulation strip force. Then, an OTPM minimum essential information analysis was used to identify all non-value added activities that resulted from poor fuselink quality. These activities were quantified in order to allow management to more accurately measure and eliminate all forms of waste involved in the fuselink production process.

6 Conclusion

Numerous field tests by practicing engineers in a variety of manufacturing companies, as well as consulting engagements conducted by the author have all shown that an OTPM-based IPFA analysis of any complex, repetitive, real-time, object transformation process can identify minimum, necessary and sufficient embedded computer information requirements. Since these requirements are in the form of well-defined input data and output information from and to people, machines, and other embedded computers in a bounded embedded system, their specificity, in addition to process timing requirements, provides a complete specification for developing process control software. From an industrial engineering perspective, when the control loops are completed and put into operation, the underlying embedded computer software can be easily validated by engineering inspection using the same IPFA methodology.

7 Acknowledgement

The author would like to acknowledge all of his experienced engineering graduate students who have carried out field tests of the OTPM-based methods described in this paper, and also the numerous manufacturing managers and information system software engineers whose helpful comments freely offered in a variety of conferences and workshops helped fine-tune the methodology over the past seven years.

8 About the Author

Dr. Manley is Professor of Industrial Engineering and Director, M.S. Degree Program in Manufacturing Systems Engineering (MSEP), University of Pittsburgh; Member of the Board of Directors Emeritus, Concurrent Technologies Corporation, Chairman, Computing Technology Transition, Inc., and LtCol Regular Air Force, Retired. His prior leadership positions include: Assistant to the Director, The Johns Hopkins University Applied Physics Laboratory; Director, Programming Applied Technology, ITT Corporation; Vice President Engineering and Technology, Nastec Corporation; and Director, Software Engineering Institute, Carnegie Mellon University.

References

1 Manley, John H., "Embedded Systems," in *Encyclopedia of Software Engineering*, John J. Marciniak, Ed., Wiley-Interscience, John Wiley & Sons, Inc., New York, 1994, pp. 454-458.

2 Manley, John H., "Managing Software in the Automated Factory," Chapter in *The Automated Factory Handbook: Technology and Management*, David I Cleland and Bopaya Bidanda, eds., TAB Professional Books, New York, 1990, pp. 198-219.

3 Manley, John H., "OTPM and the New Manufacturing Paradigm," in *Computers and Industrial Engineering*, Vol. 23, Nos. 1-4, Pergamon Press Ltd, Oxford, England, 1992, pp. 427-430.

4 Manley, John H., "Information Process Flow Analysis (IPFA) for Reengineering Manufacturing Systems," in *Computers and Industrial Engineering*, Vol. 25, Nos. 1-4, 1993, pp. 275-278.

5 Ross, D. T, Structured Analysis: A Language for Communicating Ideas, *IEEE Transactions on Software Engineering*, Vol. 3, No. 1, 1977, pp 16-33.

6 Rasmus, Daniel, "Redesigning the Corporation with IDEF's Help, *Manufacturing Systems*, December 1988, pp. 26-31.

7 Leveson, Nancy G., et al, "Requirements Specification for Process-Control Systems," *Information and Computer Science Technical Report 92-106*, University of California - Irvine, November 10, 1992, 54 pp.

8 Rine, David C. and Bharat Bhargava, "Object-Oriented Computing," *Computer*, Vol. 25, No. 10, Oct 1992, pp. 6-10.

9 Ibid, p. 17

10 Manley, John H., *Rise Above the Rest: The Power of Superior Information, Knowledge and Wisdom, Third Edition*, Cathedral Publishing, Pittsburgh, Pennsylvania, 1998.

11 Manley, John H., "Information Process Flow Analysis (IPFA) for Reengineering Manufacturing Systems," Op. Cit.